LỜI MỞ ĐẦU

Năm 2018, sự phát triển của đất nước sự hội nhập của thế giới và sự phát triển của các ngành khoa học, công nghệ thông tin ngày càng phát triển… mọi thứ hiện nay đều sử dụng tài liệu tiếng anh.Trong hệ thông giáo dục của Việt Nam hầu hết các trường tiểu học , trung học, phổ thông kể cả cấp bậc cao đẳng đại học đều đào tạo bộ môn: Tiếng Anh

Với mục tiêu nhầm hổ trợ cho việc học tập cho sinh viên , giảng dạy của giáo viên ở các trường, hổ trợ việc biên dịch tài liệu cho các ngành khoa học, công nghệ thông tin, các cơ quan chức năng,… Để không lãng phí thời gian và tiết kiệm công sức ngồi tra từng từ trong một cuốn từ điển, với sự phát triển của ngành công nghệ thông tin các phần mềm từ điển Anh-Việt xuất hiện ngày càng nhiều nhằm phục vụ cho công việc phục vụ nhu cầu của xã hội hiện nay.

Với mong muốn nhằm góp phần trợ giúp cho nhu cầu đó để phục vụ cho nền giáo dục nước nhà. Dựa trên những kiến thức tích lũy được trong khi học bộ môn: Cấu trúc dữ liệu và giải thuật, chúng em xin thực hiện đề tài: “Thiết kế từ điển Anh-Việt bằng cây tìm kiếm nhị phân” bằng ngôn ngữ Visual studio C# trên nền Windows Forms Applocation.

NHÓM THỰC HIỆN

LỜI CẢM ƠN

Chúng em xin chân thành cảm ơn thầy -Trần Công Tú :Giáo viên trực tiếp giảng dạy bộ môn “Cấu trúc dữ liệu và giải thuật” thuộc khoa công nghệ thông tin, người đã tận tình hướng dẫn chỉ dạy chúng em trong suốt thời gian học môn này và thực hiện đề tài nay.

Một lần nữa chúng em xin chân thành cảm ơn!

Nhóm Thực Hiện

**MỤC LỤC**

**[I.GIỚI THIỆU SƠ LƯỢC VỀ ĐỀ TÀI.](#_Toc28451_WPSOffice_Level1)** **[5](#_Toc28451_WPSOffice_Level1)**

[1. Giới thiệu:](#_Toc7194_WPSOffice_Level2) [5](#_Toc7194_WPSOffice_Level2)

[1. Mục tiêu:](#_Toc8669_WPSOffice_Level2) [5](#_Toc8669_WPSOffice_Level2)

[2. Các chức năng chính:](#_Toc27044_WPSOffice_Level2) [5](#_Toc27044_WPSOffice_Level2)

**[II. PHÂN TÍCH YÊU CẦU.](#_Toc7194_WPSOffice_Level1)** **[6](#_Toc7194_WPSOffice_Level1)**

**[III.CÁC BƯỚC THỰC HIỆN CHƯƠNG TRÌNH](#_Toc8669_WPSOffice_Level1)** **[6](#_Toc8669_WPSOffice_Level1)**

**[1.File từ điển txt.](#_Toc27044_WPSOffice_Level1)** **[6](#_Toc27044_WPSOffice_Level1)**

**[2. Kiến trúc chương trình.](#_Toc26252_WPSOffice_Level1)** **[8](#_Toc26252_WPSOffice_Level1)**

2.1.Mô hình hoạt động của chương trình 8

[2.2.Mô tả và giải thích xử lý các lớp, đối tượng phục vụ cho hoạt động của chương trình.](#_Toc26252_WPSOffice_Level2) [8](#_Toc26252_WPSOffice_Level2)

2.2.1.Khai báo phương thức cây nhị phân, các hàm và chức năng xử lý trong cây(tra,thêm,xóa) 8

2.2.2.Đọc dữ liệu form vào các textbox 9

2.2.3.Tìm kiếm dữ liệu (Tra từ nút button) 9

2.2.4.Chức năng thêm từ (button) 10

2.2.5.Chức năng xóa từ (button) 11

2.2.6.Chức năng đọc từ(button) 12

**[3.Các thiết kế giao diện chương trình](#_Toc5324_WPSOffice_Level1)** **[13](#_Toc5324_WPSOffice_Level1)**

[3.1.Giao diện khởi động chương trình](#_Toc30222_WPSOffice_Level2) [13](#_Toc30222_WPSOffice_Level2)

[3.2.Giao diện tra từ](#_Toc15731_WPSOffice_Level2) [13](#_Toc15731_WPSOffice_Level2)

[3.3.Giao diện thêm từ](#_Toc22583_WPSOffice_Level2) [13](#_Toc22583_WPSOffice_Level2)

[3.4.Giao diện đăng nhập để xóa từ](#_Toc24004_WPSOffice_Level2) [13](#_Toc24004_WPSOffice_Level2)

**[4.Cài đặt và thử nghiệm](#_Toc30222_WPSOffice_Level1)** **[13](#_Toc30222_WPSOffice_Level1)**

**IV.KẾT LUẬN 14**

**1.Những khó khăn và thuận lợi khi thực hiện chương trình 15**

**2.Ưu điểm, nhược điểm 16**

**3.Ý tưởng thực hiện 17**

**4.Phân công thực hiện 18**

**5.Tổng kết 19**

**6.Tài liệu kham khảo 20**

I.GIỚI THIỆU SƠ LƯỢC VỀ ĐỀ TÀI.

*1. Giới thiệu:*

Ngày nay, tiếng Anh là một ngôn ngữ rất thông dụng, được áp dụng trong mọi lĩnh vực ngày nay. Từ thức tế nhóm em đã chọn đề tài : “Thiết kế từ điển Anh-Việt” do kiến thức còn hạn chế nên chúng em đã thiết kế cho từ điển Anh-Việt những thao tác cơ bản như: tra cứu từ ngữ, thêm từ mới vào từ điển, xóa từ cần xóa. Một giao thiết kế từ điền gọn nhẹ không tiêu tốn nhiều dung lượng của bộ nhớ vẫn hổ trợ được cho công việc tra cứu làm việc và học tập một cách tiện lợi nhất.

Đề tài của nhóm lấy tên là:“DĐ-UTE Dictionary”, với ý nghĩa của từ điển(Dictionary) được thiết kế bởi nhóm hai thành viên (Danh-Đức) học tại trường Đại học sư phạm kỹ thuật ( University infomation Technology).

1. *Mục tiêu:*

Nhóm chúng em xây dựng đề tài này với mong muốn tạo ra chương trình từ điển có nguồn file txt phong phú, các tính năng cơ bản, thao tác đơn giản, giao diện đẹp, hạn chế các thao tác dư thừa dữ liệu,…. Nhằm phục vụ tối đa cho những người có nhu cầu tra cứu từ vựng và lưu những từ ngữ cần thiết vào từ điển một cách nhanh chóng.

1. *Các chức năng chính:*

-Tra từ: bằng cách nhập từ cần tra vào ô tìm kiếm hổ trợ tra cứu ngôn ngữ Anh-Việt.

-Lưu những từ mà người dùng cần lưu vào file txt để tiện sử dụng khi cần, dễ dàng cho việc học từ vựng, lưu xong người dùng cảm thấy không cần thiết nữa có thể xóa bỏ.

-Có thể thêm, xóa, từ thêm vào.

-Chức năng đọc từ (đọc bất cứ thứ gì nhập vào ô textbox tra từ sẽ phát đọc thành tiếng anh)

-Giao diện đẹp tiện dụng cho người dùng, người dùng có thể thay đổi giao diện nếu không thích giao diện hiện hành

**II. PHÂN TÍCH YÊU CẦU.**

Tuy đề tài là do nhóm chung em tự chọn và thầy không có yêu cầu gì nhưng chúng em đã tự đặt ra yêu cầu với nhóm như sau:

-Với đồ án từ điển thì việc tìm kiếm trên file txt nhanh và chính xác rất là quan trọng do vậy đòi hỏi phải sắp xếp, tối ưu hóa file txt và các thuật toán tìm kiếm cần phải tối ưu

-Giao diện đẹp, tiện dụng và có thể thay đổi nếu ngươi dùng không thích giao diện hiện hành

-Chương trình phải chạy ổn định và chức năng hoàn thiện

**III.CÁC BƯỚC THỰC HIỆN CHƯƠNG TRÌNH**

*1.File từ điển txt.*

Những cách tổ chức dữ liệu cho từ điển:

-Tổ chức dữ liệu theo dạng B-cây.

-Tổ chức dữ liệu theo dạng cây nhị phân.

-Tổ chức dữ liệu theo dạng danh sách đặc.

-Tổ chức dữ liệu theo dạng danh sách liên kết.

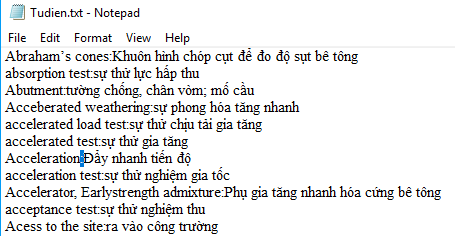
Việc chọn cách thức để tổ chức dữ liệu cho từ điển cũng khá quan trọng, và hơn hết là làm sao tổ chức thuận lợi cho quá trình tìm kiếm. Do đó nhóm đã chọn cách tổ chức theo dạng cây nhị phân.

Tổng quan file txt:

-File .txt là 1 định dạng tập tin văn bản (text file) đơn giản – không sử dụng các định dạng như chữ đậm, chữ nghiêng, màu sắc… để trình bày. Định dạng văn này được gọi là Plain Text.

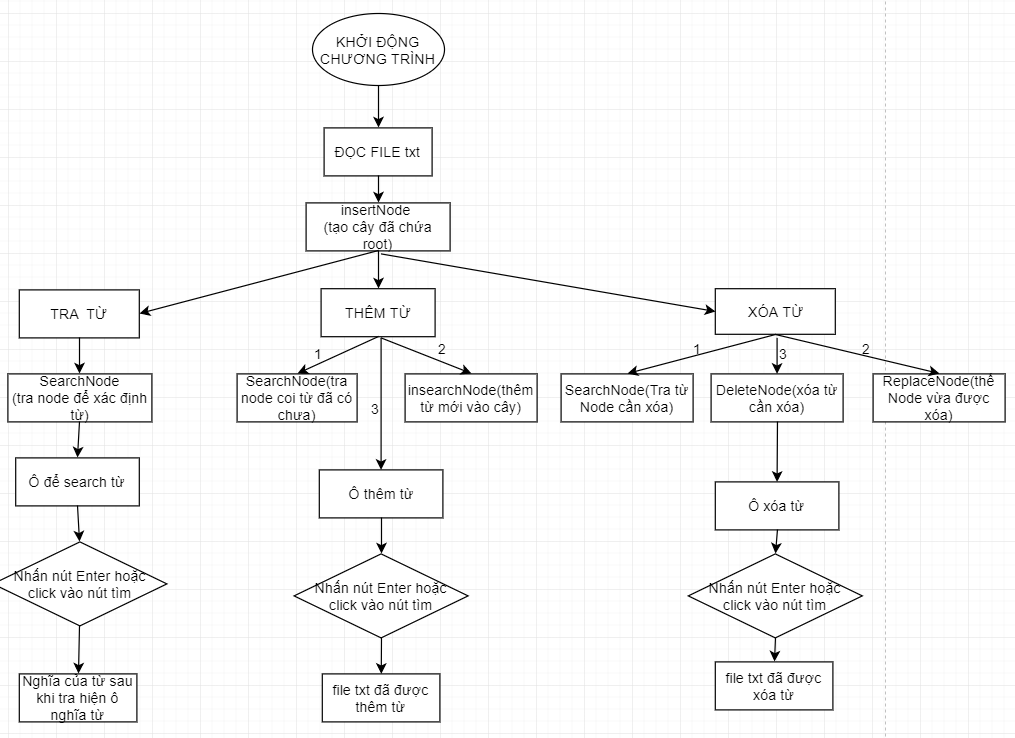
-Do từ điển Anh-Việt sẽ sử dụng so sánh mã hóa ASCII để so sánh các ký tự trong từ điển phù hợp và đơn giản cho quá trình duyệt cây nhị phân.

Cách thức khởi tạo dữ liệu cho file txt từ điển:

 Hình ảnh: mô phỏng dữ liệu trong file txt

Những từ được nhập và lưu vào trong file txt được ngăn cách nhau bởi dấu “:” để chia thành hai phần. Phần trái dấu “:” là tiếng Anh bên phải dấu “:” là tiếng Việt, mục đích để tách thành 2 textbox để hiện trên màn hình winform. Và các dữ liệu được sắp xếp danh sách theo thứ tự ABC Ascending(tăng dần).

1. *Kiến trúc chương trình.*

***2.2.1.Mô hình hoạt động chính của chương trình.***

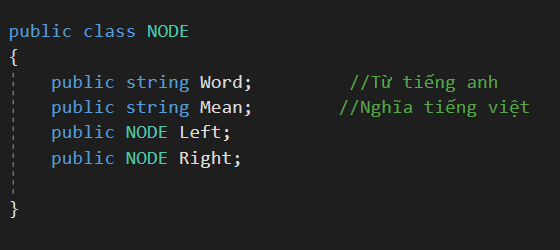
*2.2.Mô tả và giải thích xử lý các lớp, đối tượng phục vụ cho hoạt động của chương trình.*

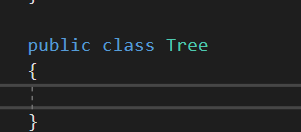
***2.2.1.Khai báo phương thức cây nhị phân, các hàm và chức năng hàm có trong cây nhị phân.***

-Ta có một class struct.cs dùng để định nghĩa và khai báo cấu trúc cho cây nhị phân.

-Khai báo Node cho cây nhị phân: Trong đó có các biến Word,Mean và nút trái, nút phải của của một Node trong cây nhị phân

-Để triển khai chường trình bằng cây nhị phân thì trong lớp struct.cs cần phải khởi tạo một cây trong đó chưa tất cả các hình thức hoạt động của cây.

 Hình ảnh khai báo Node



Hình ảnh khởi tạo cây

-Và đây là những hàm cần có trong cây để làm 1 từ điển Anh-Việt:các biến được khai báo truyền vào trong hàm

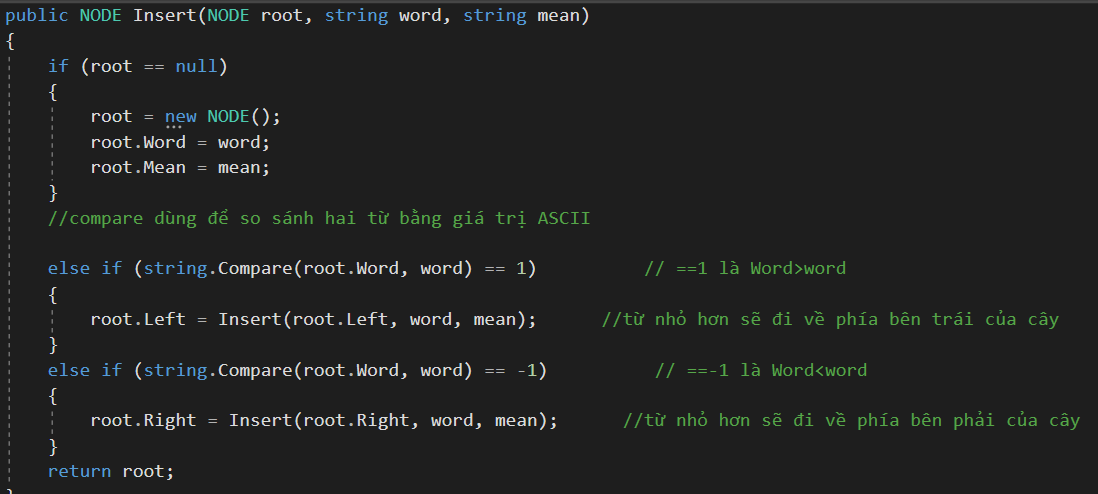
○Hàm NODE Insert:

◦ NODE root

◦ word: từ tiếng Anh do người dùng nhập vào

◦mean: nghĩa tiếng Việt do người dùng nhập vào

Hàm NODE Insert dùng xác định root và có chức năng chèn thực hiện để bổ trợ cho các lớp khác. Ở đây ta thấy root==null là trường hợp không có root nào cả nên ta tạo một root=new NODE(), truyền vào cho nó từ tiếng Anh và nghĩa của từ đó, ngược lại dùng Chuổi.Phương thức so sánh bằng hàm Compare: so sánh hai chuổi ký tự bằng giá trị ASCII. Ở đây so sánh root.Word với word (từ tiếng có trong file txt với từ tiếng Anh người dùng thêm vào) nếu bằng 1 thì sẽ chèn từ đó vào phía bên phải của cây, ngược lại thì chèn vào bên trái của cây, và ngược lại tiếp tục thì reutn root phương thức gọi đệ quy để quay lại tìm kiếm hoạt động lặp đi lặp lại cho đến hết cây nhị phân.



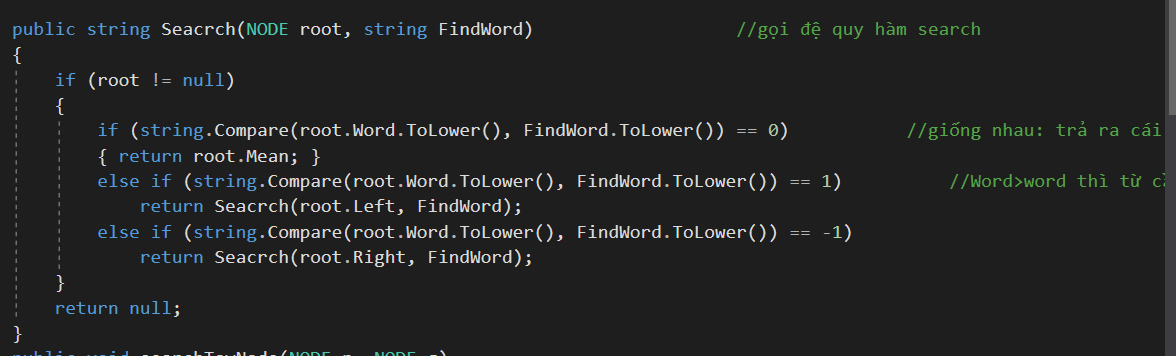
Hình ảnh hàm Insert trong lớp Tree

○Hàm Search

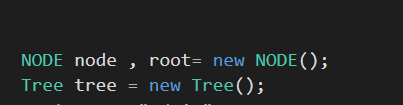
◦NODE root

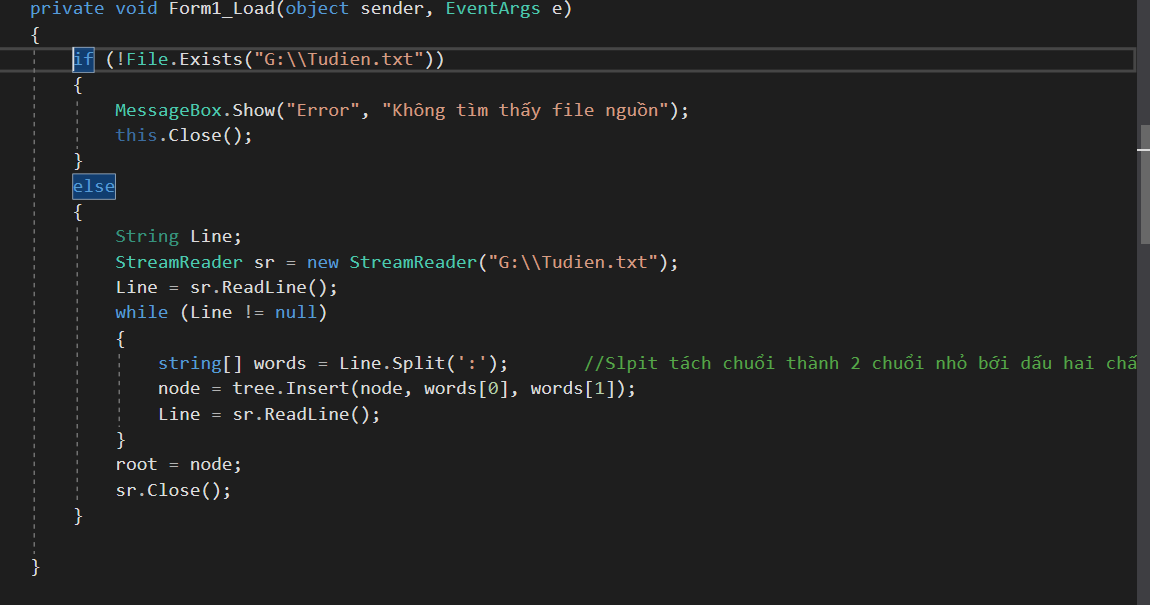
◦FindWord: người dùng nhập vào từ cần tra

Ở đây ta bắt đầu gọi đệ quy hàm search. Cũng truyền vào root, nhưng ở đây ta có FindWord để tra từ, ở đây root!=Null là cây có từ thực hiện các phương thức trong if tiếp tục ta sử dụng Chuổi.phương thức so sánh bằng hàm Compare(), ta thấy có thêm .ToLower() có chức năng là không phân biệt chứ hoa và chữ thường giữa các kí tự. Khi root.Word từ có trong file txt = với từ người dùng nhập vào FindWord (==0) thì return trả ra root.Mean nghĩa của từ đó, ngược lại Word>FindWord thì từ đó sẽ chạy về phía bên trai ngược lại chạy về phía bên phải, return null gọi đệ quy để chạy thực thi lại khi đến khi chạy hết cây hoặc hai từ đó bằng nhau.

Hình ảnh hàm Search trong lớp Tree

***2.2.2.Đọc dữ liệu vào form cho các textbox(Đang lũng củng)***



Hình ảnh load dữ liệu lên các text box

Xét hàm void Form1\_load():

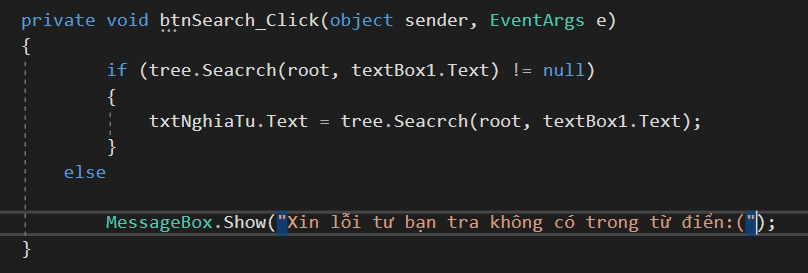
-Ta thấy có biến string Line; được định nghĩa là 1 dòng từ và nghĩa trong file txt từ điển.

-Đối với C# không cần phải suy nghĩ làm sao để có thể đọc được nội dung từ 1 file txt. Ta chỉ cần thêm thư viện using System.IO;(sử dụng thư viện này để làm việc với StramReader). Việc sử dụng rất đơn giản khởi tạo một đối tượng cho StreamReader và truyền vào 1 đường chỉ dẫn đến nơi lưu file cần đọc trong máy tính.

-sr.Readline(): đọc dữ liệu txt và đưa vào Line (ReadLine: là đọc xong hàng xuống dòng đọc tiếp đọc đến hết file txt thì dừng lại). Line được kiểm tra trong vòng lặp While(Line==Null) thì root=node và sẽ đóng đối tượng close(), Khi Line!=Null thì những câu lệnh bênh trong sẽ được thi hành do file dữ liệu từ điển txt như trên nên được tách thành hai chuổi khác nhau: tiếng Anh, tiếng Việt

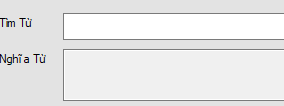
***2.2.3.Tìm kiếm dữ liệu (tra từ nút button)***





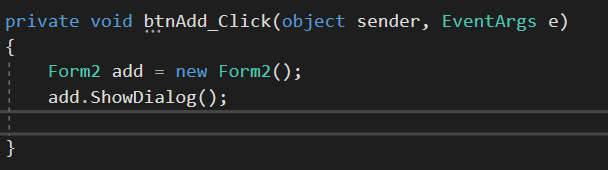
Hình ảnh nút button và phương thức sử lý

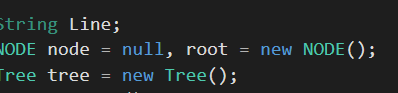
Nút button được bắt sự kiện bằng double click phương thức sử lý của nút, gọi cây và truy xuất các thành phần của search ở đây hàm search đã được thực thi trong lớp Tree. Ở đây nút search sẽ hoạt động như sau: ở đây có một cây được truy xuất các thành phần search root giá trị được trả về khi hàm search so sánh với texbox1.Text là ô từ để nhập từ nếu ở đó khác null chứng tỏ người dùng đã truyền vào từ cũng được hiểu như FindWord trong hàm search, thực hiện các bước so sánh trong search khi tìm được từ đó thì nghĩa từ đó được trả ra vào textbox được đặt tên là txtNgiaTu và ngược lại nếu không tìm thấy thì xuất ra màn hình “xin lỗi từ bạn tra không có trong từ điển”



Hình ảnh về các textbox

***2.2.4.Chức năng thêm từ(button)***

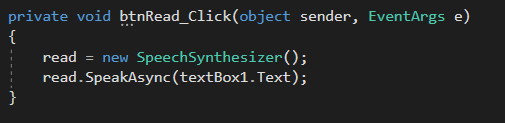




***2.25.Chức năng xóa từ(button)***

***2.2.6.Chức năng đọc từ(button)***



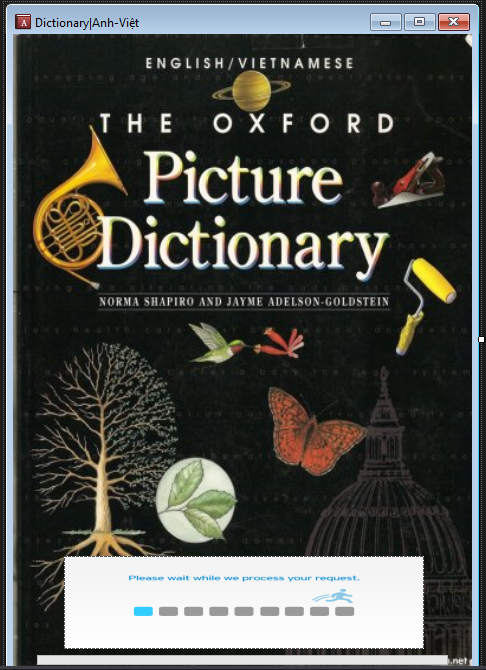
Hình ảnh nút button giọng nói

Để sử dụng chức năng này ta cần phải khai báo thư viện:using System.Speech.Synthesis, cung cấp quyền truy cập vào chức năng của một công cụ tổng hợp giọng nói được cài đặt. Khai báo một biến Read được khởi tạo biến Read và dùng SpeakAsync được hổ trợ bởi thư viện trên: cung cấp các phương tiện đơn giản nhất để tạo ra đầu ra lời nói

***3.Các thiết kế giao diện chương trình***

***3.1.Giao diện khởi động chương trình***

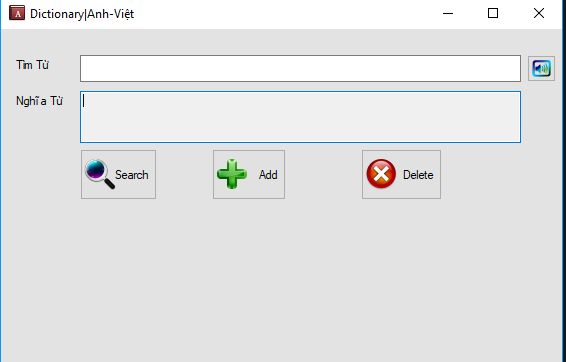
Form4 (text: Dictionary|Anh-Việt)



Khi khởi động chương trình từ điển, hiệu ứng loading và hình nền giúp cảm giác thú vị hứng thú khi mở từ điển để chuẩn bị cho giờ làm việc của mình.

***3.2.Giao diện tra từ***

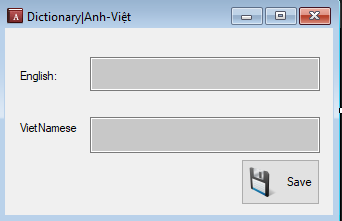
Form1(text:Dictionary|Anh-Việt)



Nhằm mục đích tiện lợi và nhanh chóng nhất để cho người dùng có thể tra cứu từ nghữ , vì do file dữ liệu txt từ tiếng anh được cấu trúc Anh:Việt, nên đã tách thành Tìm từ dùng để nhập từ tiếng anh vào texbox kế bên, nghĩa từ xuất nghĩa ra textbox kế bên nó. Ở nút search có hình kính lúp biểu tượng cho sự tìm kiếm có kèm tiếng Anh, nhằm tạo sự bắt mắt với những khung textbox cơ bản giúp cho người dùng dễ sử dụng.

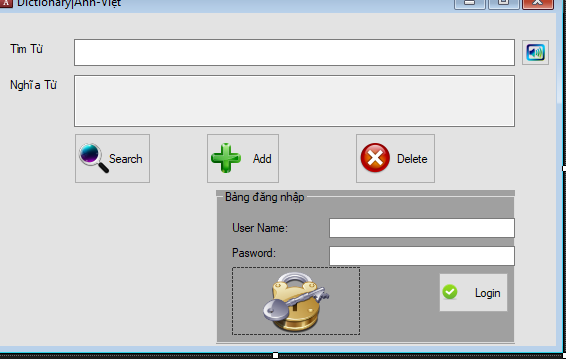
***3.3.Giao diện thêm từ***

Form2(Dictionary|Anh-Việt)



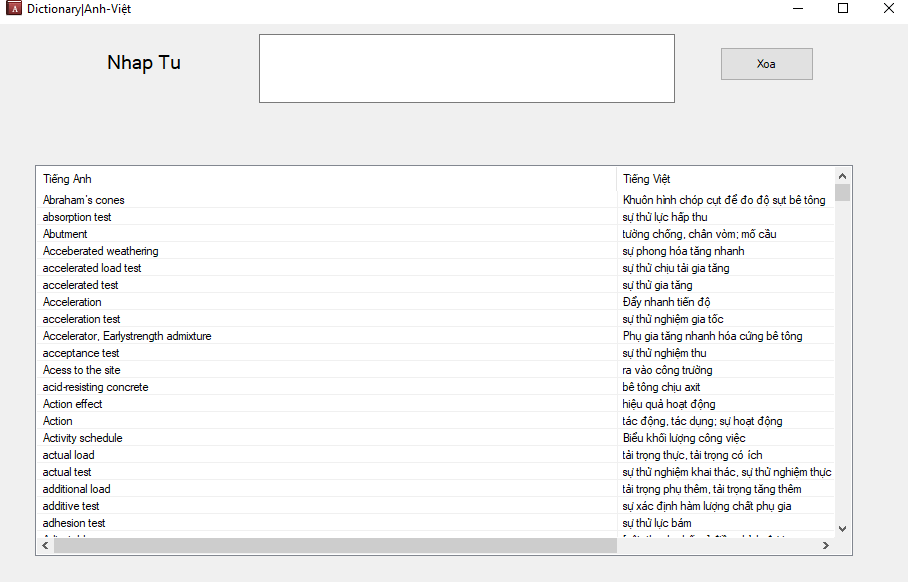
Bởi cấu trúc dữ liệu được cách nhau bởi “:” nên khi lưu từ vào file txt cũng phải thiết kế 2 textbox nhầm có thể lưu từ vào file txt. Mục đích dùng để thêm từ mới từ người dùng muốn thêm từ vào txt. Khi nhấn nút save thì từ sẽ được thêm vào mới từ bị trùng hoặc đã có trong từ điển thì sẽ báo ra màn hình từ đã có mời bạn nhập lại. Form2 này sẽ hiện lên khi được lick chuột vào nút button Add bên Form1 tức khắc sẽ nhảy vào Form2 này thực hiện quá trình lưu.

***3.4.Giao diện đăng nhập để xóa từ***



Bảng đăng nhập này được load cùng với Form1 nhưng sẽ được ẩn đi khi chương trình được chạy và khi nút button Delete được lick thì Bảng đăng nhập sẽ được hiện ra và ta cần phải nhập User, Password để có thể truy xuất vào Form3 để xóa từ. Vì tính năng xóa từ ảnh hưởng đến sự hao hụt của dữ liệu nên cần bước đăng nhập để bảo toàn thông tin dữ liệu đã được lưu trong file txt.( Bảng đã được làm ẩn bằng thuộc tính Visible chế độ False. Có hình ảnh làm bắt mắt người sử dụng .

***3.4.1.Giao diện xóa từ***



Được thiết kế nhằm có thể tìm từ một cách nhanh chóng và có một list danh sách từ ngữ bạn có thể xem và kiểm tra có từ nào sai cần xóa chỉ cần thao tác click chọn một cách đơn giản.

***4.Cài đặt và thử nghiệm***

**IV.KẾT LUẬN**

1. Những khó khăn và thuận lợi khi thực hiện chương trình

-Thuận lợi:

◦Dữ liệu có sẵn trên mạng nên không phải tự tay tạo ra nguồn database có tư điển.

-Khó khăn:

◦Do lấy dữ liệu free trên mạng nên gặp rất nhiều khó khăn để nghĩ ra phương thức để tách và đọc dữ liệu cộng với lần đầu tiên làm một đồ án cùng với ngôn ngữ mới C# nên gặp nhiều khó khăn trong quá trình tìm hiểu và bắt tay vào viết chương trình. Nhưng nhóm đã cố gắng thực hiện đồ án của mình tuy đã có trễ hơn thời gian dự định nhưng đồ án vẫn thành công đúng thời hạn.

2.Ưu điểm và nhược điểm

-Ưu điểm:

◦Giao diện đẹp, tiện dụng.

◦Khối lượng từ tương đối đầy đủ.

◦Có âm thanh giúp người đọc nghe được từ vựng.

◦Có chức năng lưu từ giúp người học từ vựng dễ dàng hơn.

-Nhược điểm:

◦Chu trình hoạt động của từ điển còn gặp nhiều rắc rối.

◦Còn thiếu sót nhiều chức năng khác nhau.

◦Đòi hỏi phải có 1file txt gắn liền với từ điển. Khi di chuyển từ điển từ máy này qua máy khác file txt có thể bị thay đổi ổ lưu có thể từ điển sẽ không đọc được file.

3.Ý tưởng thực hiện

4.Phân công thực hiện

5.Tổng kết

6.Tài liệu kham khảo